2D Adaptive Grid-based Image Analysis Approach for Biological Networks

Haifa F. Alhasson
School of Engineering and Computing Science,
Durham University, Durham, UK.
Email: h.f.alhasson@durham.ac.uk

Boguslaw Obara
School of Engineering and Computing Science,
Durham University, Durham, UK.
Email: boguslaw.obara@durham.ac.uk

***Abstract*—The accurate analysis of biological networks, enabled by the precise capture of their individual components, can reveal important underlying biological principles. Efficient image processing techniques are required to precisely identify and quantify the networks from complex images. In this paper, we present a novel approach for a weighted and undirected graph-based network reconstruction and quantification from 2D images using an adaptive rectangular mesh refinement approach. The proposed approach is able to efficiently identify the organizational principles of the network, capturing the underlying network structure, and computing relevant network topological properties. We validate the proposed approach by comparing it with the state-of-the-art method.**

I. Introduction

BIOIMAGE informatics is a sub-field of bioinformatics and computational biology that focuses on management, processing, and quantification of biological structures in the images. Particularly, biological networks widely studied by the bioimage informatics community [1], [2]. However, information of the content and complexity of biological networks in images vary significantly between different network subcategories, such as, the very high density of links in 2D leaf vein networks [3]. Therefore, there is an increasing need to develop an image processing approach in conjunction with biologists, combining the collaborative knowledge of biology, computer science, mathematics, and other related disciplines in order to provide the most effective ways for quantification of complex biological networks. Such an approach can also be applied to other types of networks, such as neural networks or transportation systems [4].

In this paper, we propose a quantitative and biologically driven image processing approach able to quantify complex biological networks, by first multiscale reconstructing the network as an undirected graph, with nodes and weighted edges, and then quantifying the network's topological properties.

II. Related Work

A. Direct network reconstruction

A direct reconstruction of biological networks relies on image segmentation techniques. For example, in [5], vein networks were analysed using thresholding-based segmentation followed by multiple restoration algorithms. Then, the skeleton for the segmented image is computed, followed by some correction steps. Finally, the image skeleton is scanned to obtain the network topology. In [2], a contrast invariant curvilinear network-like features enhancement step followed by the watershed-based network segmentation algorithm is proposed to enhance, extract and quantify the fungal, leaf vein and vascular networks. Moreover, in order to precisely determine the network's topological properties, a mathematical graph-based representation of the network is also introduced. Most recently, a novel and generic platform called Network Extraction From Images (NEFI) has been developed by [1]. The platform provides a wide range of image pre-processing and segmentation techniques for network reconstruction and graph-based representation.

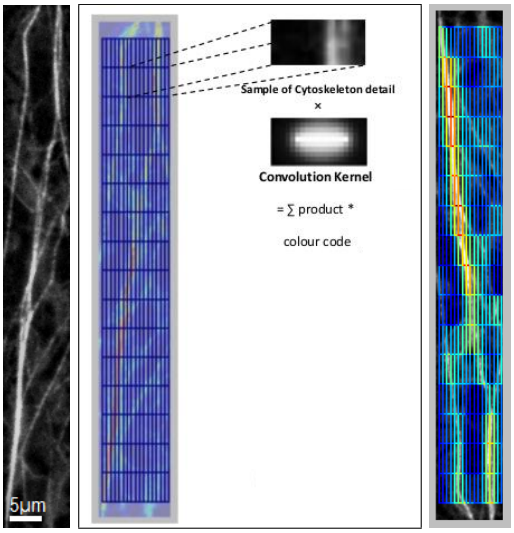
B. Indirect network reconstruction

Alternatively to the direct network reconstruction approaches, an indirect network reconstruction approach called the img2net was proposed by [4], [6]. The img2net reconstructs a network from 2D/3D image using a two-step procedure as shown in Figure 1. First, a uniform 2D/3D grid defined by $[dx, dy]$ spacing (or $[dx, dy, dz]$ in 3D) is generated over the network image. Then, a weighted graph is reconstructed, representing the positions of grid junctions as nodes and the grid edges as edges. The graph edge weights are determined by convolving the image with a set of 2D/3D Gaussian kernels. Importantly, the img2net approach is implemented using the NetworkX Python library [7] which is firmly based on graph theory concept. This therefore allows for a quick and direct use of a wide range of very sophisticated quantitative measurements of the reconstructed graph representation of the analysed network.

C. Mutliscale indirect network reconstruction

Since curvilinear network-like structures observed in the biological images can appear in different sizes, a multiscale reconstruction and quantification of the networks needs to be considered. A lack of such a multiscale concept is a main drawback of the img2net approach as the img2net is based on a regular uniform grid construction procedure. Fortunately, this drawback can be addressed by an introduction of an adaptive grid concept into the img2net approach.

One of the most common adaptive grid concepts is a structured Adaptive Mesh Refinement (AMR) [8]. The AMR is extensively used in the simulations of scientific phenomena exhibiting large variations in scales [9], by using spatial



(a) (b) (c)

Figure 1: The img2net approach workflow [6]. (a) 2D cytoskeletal network image. (b) The uniform grid construction (parameters: $gridtype = rectangular$, $nullmodel = edges$, $dx = 2.5$ pixels, $dy = 22$ pixels and $dz = 1$ pixel), and edge weights estimation. (c) The resulting network's representation by weighted and undirected graph, where the weights of edges are colour-coded from blue (low value) to red (high value).

partitioning methods to dynamically adjust the resolution of the computational mesh/grid to features in the solution or computational domain [10].

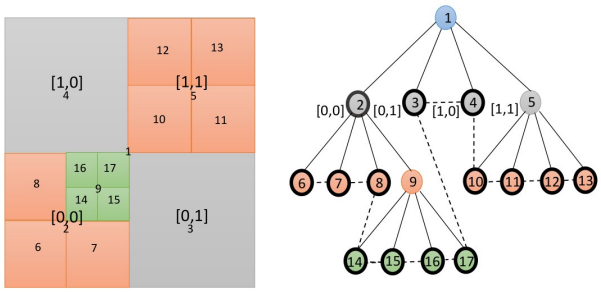
1) Quadtree:

A quadtree is a well-known tree-based data structure used in the adaptive grid concepts, where every internal node has four points and every node represents a quadrant(square or rectangle) [10]–[14]. Recursive quadtree partitioning captures the features in a computational problem domain by setting the minimum and maximum resolution of the computational grid. The recursive approach of quadrant partitioning halts at certain points, based on specified stop conditions related to the problem domain or if the number of these points is less than a specified size constraint [15].

As an example, Figure 2 illustrates the hierarchy of a sample quadtree and how the labels could be used to access the data. The dashed lines show how the nodes are accessed directly without using parent-child relation.

2) Spatial hashing:

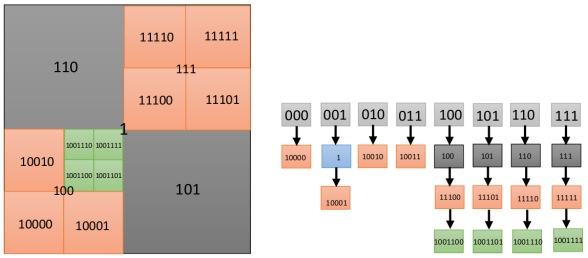
Accordingly to [16], [17], it is known that a standard pointer quadtree data structure leads to random memory access during nodes traversal. Therefore, to increase the accessibility and compactness of a quadtree data structure, pointers in index manipulation is replaced by spatial hashing technique. In such case, the quadtree data is re-organised according to a one-dimensional ordering in memory. In particular, Lefebvre [18] proposed a direct access method to any node in quadtree, by storing the index data using a small offset table at each level and encouraging each hash function to reach the minimal perfect hash function. Further work by Lewiner [17] introduced pointer-less quadtree providing faster performance with less



(a) (b)

Figure 2: The quadtree concept. (a) An example of a quadtree. (b) The structure of the tree (top-down) and labeling approach used to identify the position of the subdivided box [16].

memory consumption. This concept is shown in Figure 3.



(a) (b)

Figure 3: Pointer-less quadtree data structure representation proposed by [17]. (a) Morton Key coding that is used in blocks traversal. (b) The hierarchy of blocks that illustrates block positions (hash table (3-last bits of key: k & $\bar{0}111$)).

3) Post-refinement constrains:

Instead of using a user-defined thresholding in top-down quadtree refinement [19] or using ground truth images to test the quality of quadtree decomposition [20], we automated the quadtree-based grid construction based on image content by utilizing it to have an automated thresholding selection. A wide range of image features, including: intensity, contrast, colour, and texture can be used in refining a quadtree's quadrants. In this work, we have done a post-refinement process based on a global (Otsu [21]) or a local (Niblack [22] and Sauvola [23]) image intensity thresholding.

III. Methods

In this study, we propose a novel approach for capturing complex biological networks from 2D images. We use an Adaptive Mesh Refinement (AMR) concept in order to achieve an efficient multiscale network reconstruction and quantification. Algorithm 1 summarizes all steps of the proposed approach.

A. Pointerless-quadtree-based adaptive grid construction

1) Quadtree:

To generate the quadtree-based grid, we use a pointerless quadtree concept originally proposed by [16] which uses the hashing method proposed by [17] and shown in Figure

3 to store the quadtree nodes instead of using parent-child relation in nodes traversals in pointer quadtree. By using Depth-First Search (DFS) ordering in nodes traversals a pointerless quadtree provides slightly optimization in memory utilisation with shorter traversals.

Each quadrant q^d in the quadtree-based grid at depth d, is defined by a corresponding set of coordinates:

$$q^d = [x^d, y^d, x^d + \Delta_{x^d}, y^d + \Delta_{y^d}] \tag{1}$$

where

$$\Delta_{x^d} = \frac{w}{2^d}, \quad \Delta_{y^d} = \frac{h}{2^d} \tag{2}$$

where $w \times h$ is a grid size. At every recursive quadrant partitioning step, each quadrant q^d at depth d is recursively subdivided into its four sub-quadrants:

$$q^{d+1}_{[m,n]} = \left\{ q^{d+1}_{[0,0]}, q^{d+1}_{[0,1]}, q^{d+1}_{[1,0]}, q^{d+1}_{[1,1]} \right\} \tag{3}$$

defined by a following set of coordinates:

$$\begin{aligned}
q^{d+1}_{[0,0]} &= [x^d, y^d, x^d + \frac{\Delta_{x^d}}{2}, y^d + \frac{\Delta_{y^d}}{2}] \\
q^{d+1}_{[0,1]} &= [x^d + \frac{\Delta_{x^d}}{2}, y^d, x^d + \Delta_{x^d}, y^d + \frac{\Delta_{y^d}}{2}] \\
q^{d+1}_{[1,0]} &= [x^d, y^d + \frac{\Delta_{y^d}}{2}, x^d + \frac{\Delta_{x^d}}{2}, y^d + \Delta_{y^d}] \\
q^{d+1}_{[1,1]} &= [x^d + \frac{\Delta_{x^d}}{2}, y^d + \frac{\Delta_{x^d}}{2}, x^d + \Delta_{x^d}, y^d + \Delta_{y^d}]
\end{aligned} \tag{4}$$

All $q^d_{[m,n]}$ sub-quadrants are ordered using Morton code, as shown in Figure 3(a).

2) Spatial hashing:

This work applies the same hashing principle as in [16], [24], and [17]. We store the depth and the quadrant coordinates for each node using a hashing method, where the d is the level (depth) of the subdivision and C is a tuple, which defines the subdivision labeling of a certain image block, where C can be calculated recursively as follows:

$$C = 2 * [m^d, n^d] + [m^{d+1}, n^{d+1}] \tag{5}$$

where $[m^d, n^d]$ and $[m^{d+1}, n^{d+1}]$ are a tuples refers to subdivision labelling of quadrants q^d and q^{d+1} in a grid Q.

3) Partitioning:

Let us consider a 2D grey-scale image $I(x, y)$ with a size of $w \times h$. In order to build a quadtree on an image I the non-uniformity constraints have to be satisfied and, therefore, the minimum image block size s_{min} is a user-input with $s_{min} \geq 2$ because the minimum region of image that can partitioned is 4 pixels. In addition, the 2D quadtree-based grid generation procedure is controlled by a maximum grid partitioning depth d_{max} which can be calculated according to [25] as follows:

$$d_{max} = \log(\frac{\min(w, h)}{s_{min}}) + \frac{3}{2} \tag{6}$$

The considered quadrants in the final adaptive grid have to satisfy a weighted multi-level threshold T. In the first step, we construct the quadtree to calculate the weighted multi-level threshold where each region of the image defined by a grid's quadrant q^d, can be denoted as $I(q^d)$. A threshold value $t^d_{q_i}$ is calculated as a minimum thresholding value for all four sub-quadrants $q^{d+1}_{[m,n]}$ as follows:

$$\begin{aligned}
t^d_{q_i} &= \min \left\{ T(I(q^{d+1})) \right\} \\
&= \min \left\{ T(I(q^{d+1}_{[0,0]})), T(I(q^{d+1}_{[0,1]})), T(I(q^{d+1}_{[1,0]})), T(I(q^{d+1}_{[1,1]})) \right\}
\end{aligned} \tag{7}$$

For every depth d, the average depth thresholding is calculated t^d_{avg} in the following:

$$t^d_{avg} = \frac{\sum_{i=1}^{2^d} t^d_{q_i}}{2^d} \tag{8}$$

From Equation 8, a weighted multi-level threshold is calculated using the global threshold $t = T(I)$ as follows:

$$t^d_{avg} = \frac{t^d_{avg} + t}{2} \tag{9}$$

In order perform a quadtree refinement, we assume that the grid's quadrant can be considered only if the following thresholding condition is satisfied:

$$t^d \geq \frac{{t^d_{avg}}^2}{t} \tag{10}$$

The condition in Equation 10 is inferred from the relation between dividing the current quadrant's thresholding t^d by the average thresholding t^d_{avg} compared with the average thresholding t^d_{avg} divided by the global thresholding of an image I as follows:

$$\frac{t^d}{t^d_{avg}} \geq \frac{t^d_{avg}}{t} \tag{11}$$

To test the proposed quadrant partitioning procedure, Otsu (global), Sauvola's and Niblack's (local) image thresholding approaches were used. For each thresholding method, mean and standard deviation of image intensity are calculated within image region $I(q^d)$ defined by a corresponding quadrant q^d.

4) Graph edge weights:

For each edge of the quadrant in the last depth $q^{d_{max}}$, the edge weight is defined by the convolution of an image I, at the edge position, with a Gaussian kernel of a size correlated with the last depth of quadrant partitioning process based on using the optimal sigma size in [26] for our target image . This procedure is illustrated in Figure 4.

B. Network graph-based analysis

After building the adaptive grid-based graph representation of the network, we analyse the network properties in a way similar to [6]; see Table IV for more details.

IV. Results

A. AG-img2net vs. img2net approach

In order to clearly show the effectiveness of the adaptive grid concept (AG-img2net), we have chosen different types of networks in order to reconstruct and quantify their topology,

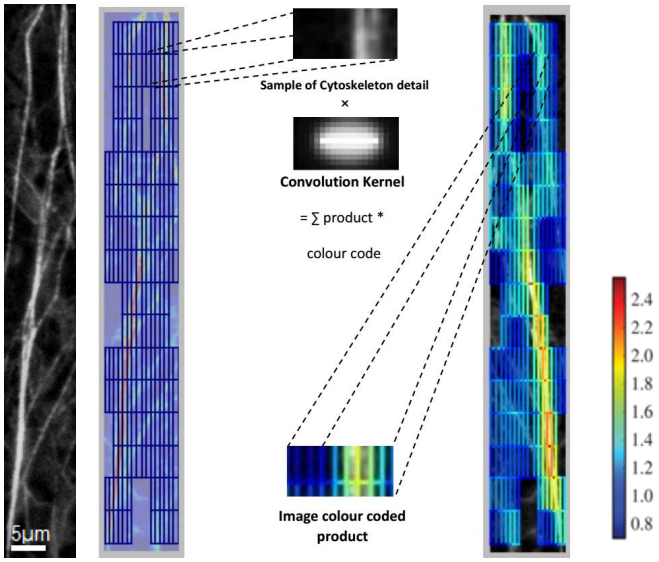


Figure 4: Reconstruction of the network using a quadtree concept. (a) 2D grey-scale cytoskeletal image. (b) The constructed adaptive grid using a quadrant partitioning procedure based on Otsu's thresholding method ($s_{min} = 4$ pixels and $k = 1$). (c) The resulted network's representation of weighted and undirected graph, where the weights of graph's edges are colour-coded from blue (low value) to red (high value).

including synthetic networks and real networks such as cytoskeleton in plant cells used in img2net [4], [6] and keratin in skin cells.

1) Refined/unrefined AG-img2net vs. img2net - synthetic image: Figure 5(a) shows a placed grid on a synthetic network image with 2500 grid junctions and 4900 edges using the unconditional version of our approach and the img2net [6] approach, therefore they have similar resulting network graphs illustrated in Figure 5(b). Table I illustrates a comparison between properties of the reconstructed networks, where the standard deviation of node degree, degree assortativity, algebraic connectivity and the average path length are almost the same for all tests. The test on the same image using a conditional version of our approach shows that the underlying network can be captured accurately with fewer networks entities (838 grid junctions and 3339 edges) as shown in Figure 5(c), and 5(d). The difference between value of the reconstructed network's properties refers to a slight fractional difference in the values of grid's positions between img2net and AG-img2net.

2) Unrefined AG-img2net vs. img2net - cytoskeleton image: The AG-img2net (with no refining) and img2net were applied to the same cytoskeleton image used in [6] and in Figure 1. And then, a few major topology measurements of the network were compared, including standard deviation of node degree, degree assortativity and algebraic connectivity of the network, see Table I,

Algorithm 1: 2D quadtree-based grid generation algorithm

Input: I, s_{min}
Output: Q

$[w, h] \leftarrow size(I)$
$d_{max} \leftarrow \log(\frac{\min(w,h)}{s_{min}}) + \frac{3}{2}$
$d \leftarrow 0$
$\Delta_{x^d} \leftarrow \frac{w}{2^d}$
$\Delta_{y^d} \leftarrow \frac{h}{2^d}$
$x^d \leftarrow 0$
$y^d \leftarrow 0$
$q^d \leftarrow [x^d, y^d, x^d + \Delta_{x^d}, y^d + \Delta_{y^d}]$
$Q \leftarrow \{q^d, d\}$
$t^d_{avg} \leftarrow 0$
while $(min(\Delta_{x^d}, \Delta_{y^d}) > s_{min}) \wedge (d \leq d_{max})$ **do**
  $q^{d+1}_{[m,n]} \leftarrow split(q^d)$
  $Q \leftarrow \left\{Q, \left\{q^{d+1}_{[m,n]}, d+1\right\}\right\}$
  if $d < d_{max}$ **then**
    $t^d \leftarrow \min\left\{T(I(q^{d+1}_{[m,n]}))\right\}$
    $t^d_{avg} \leftarrow t^d_{avg} + t^d$
  end if
  $d \leftarrow d + 1$
  $\Delta_{x^{d+1}} \leftarrow \frac{w}{2^{d+1}}$
  $\Delta_{y^{d+1}} \leftarrow \frac{h}{2^{d+1}}$
end while
for $i = 0$ to d_{max} **do**
  if $i \neq d_{max}$ **then**
    $t \leftarrow t^i_{avg}$
  else
    $t \leftarrow t^{i-1}_{avg}$
  end if
  if $t(q^i) < \frac{{t^i_{avg}}^2}{T(I)}$ **then**
    $Q \leftarrow \{Q, delete\ q^d\}$
  end if
end for
return Q

B. Refining the quadtree-based grid in AG-img2net - using the traditional thresholding vs. multi-level thresholding

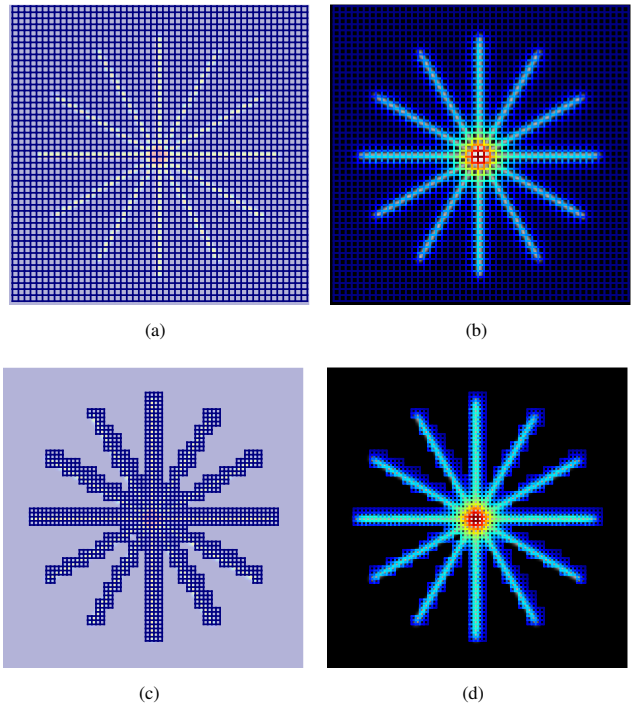
Figure 6 shows refining the quadtree-based grid in AG-img2net using the traditional thresholding and our multi-level thresholding.

V. Validation

In order to validate our approach, we test it from two perspectives: the relevancy of measured network properties to it's biological behaviour and performance of the approach.

A. Validating the biological relevancy of measured properties

First, we test whether the AG-img2net identifies biologically meaningful properties by analysing cytoskeleton images analysed before by img2net and presented in [4], [6]. To do



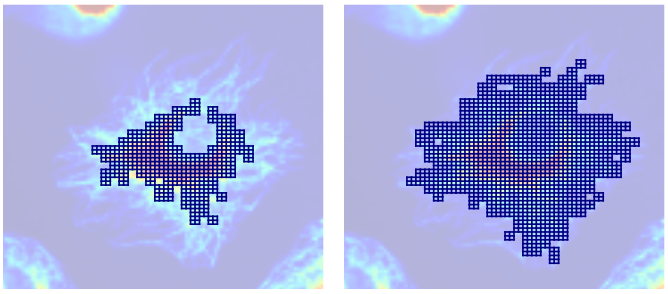
(a) (b) (c) (d)

Figure 5: Comparison between AG-img2net and img2net approach using a synthetic network. (a) Grid construction using AG-img2net ($s_{min} = 2$ pixels, no refining) or img2net ($dx = 2$ pixels, $dy = 2$ pixels and $dz = 1$ pixel). (b) Reconstructed weighted graph representation of the underlying network using both approaches. (c) Grid construction using AG-img2net ($s_{min} = 2$ pixels, with refining based on Sauvola's thresholding: $k = 2$ and $s_{min} = 2$). (d) Reconstructed weighted graph representation of the underlying network using AG-img2net.

Table I: Comparison between AG-img2net (with no refining) and img2net applied to the image in Figure 1.

Measurements	Image samples			
	Cytoskeleton		Synthetic	
	img2net	AG-img2net	img2net	AG-img2net
Sandard deviation of nodes degree	0.00384	0.00384	0.0177	0.0177
Degree assortativity	0.778	0.778	0.78317	0.78317
Average path length	5555.6	5560.9	4.00E+39	4.00E+39
Algerbric connectivity	0.000051	0.000051	0	0

this, we have followed the same validation procedure as in [6] by comparing 18 samples of cell cytoskeleton, including cells treated with *Latrunculin B* and untreated ones; the obtained results are presented in Figure 7. Based on the obtained results we can conclude that the cytoskeleton networks of treated cells shows a lower average node degree and deviation of the degree distribution (independent two-sample t-test for both (0.001 < 0.05)). Skewness of degree distribution is higher in treated samples than the control samples (independent two-sample t-test for both (0.0001 < 0.05)). A comparison of the degree assortativity of treated and untreated networks using our analysis approach AG-img2net shows a statistical reduction



(a) (b)

Figure 6: Comparison between the traditional thresholding and our multi-level thresholding in refining the quadtree-based grid. (a) A grid constructed by quadtree-based grid partitioning ($s_{min} = 2$ pixels) then refined using the traditional Otsu thresholding. (b) A grid constructed by quadtree-based grid partitioning ($s_{min} = 2$ pixels) then refined using a multi-level Otsu thresholding (our approach) ($k = 1$).

for cytoskeleton treated with *Latrunculin B* (independent two-sample t-test = 0.05). Both findings agree with the results using img2net approach [4], [6] (independent two-sample t-test = 0.04 < 0.05, independent two-sample t-test = 0.01 < 0.05), respectively. Moreover, the treated networks show a significant reduction in average path lengths (APL) using AG-img2net and img2net approach (independent two-sample t-test = 0.001 < 0.05, independent two-sample t-test = 0.02 < 0.05), respectively. Skewness of the shortest path lengths increased in treated samples compared to the control ones using our approach (independent two-sample t-test = 0.01 < 0.05), while it is an insignificant property using img2net approach (independent two-sample t-test = 0.9 > 0.05). In addition the radius and diameter of the network of treated cytoskeleton expanded in treated samples (independent two-sample t-test for both (0.001 < 0.05)). Algebraic connectivity of cytoskeleton images steeply declined after chemical treating (independent two-sample t-test for both (0.04 < 0.05)). Furthermore, current-flow betweenness centrality was higher in treated samples than untreated ones in AG-img2net (independent two-sample t-test = 0.009 < 0.05) while it is an insignificant property in img2net approach (independent two-sample t-test = 0.1 > 0.05).

B. Robustness and performance

Robustness of the AG-img2net approach for quantitative representation of a wide range on biological networks is shown in Table II. Moreover, computing time comparison, for corresponding image examples from Table II, using AG-img2net and img2net approaches is shown in Table III.

VI. Conclusion

In this paper, we introduced an efficient multiscale network reconstruction and quantification approach using an adaptive grid concept. First, we proposed to use an adaptive grid approach based on pointerless quadtrees, allocating network nodes more precisely than regular grids and requiring fewer grid components. Second, an automatic quadtree partitioning based on a global and local thresholding has been proposed for capturing the network bundles accurately. This approach

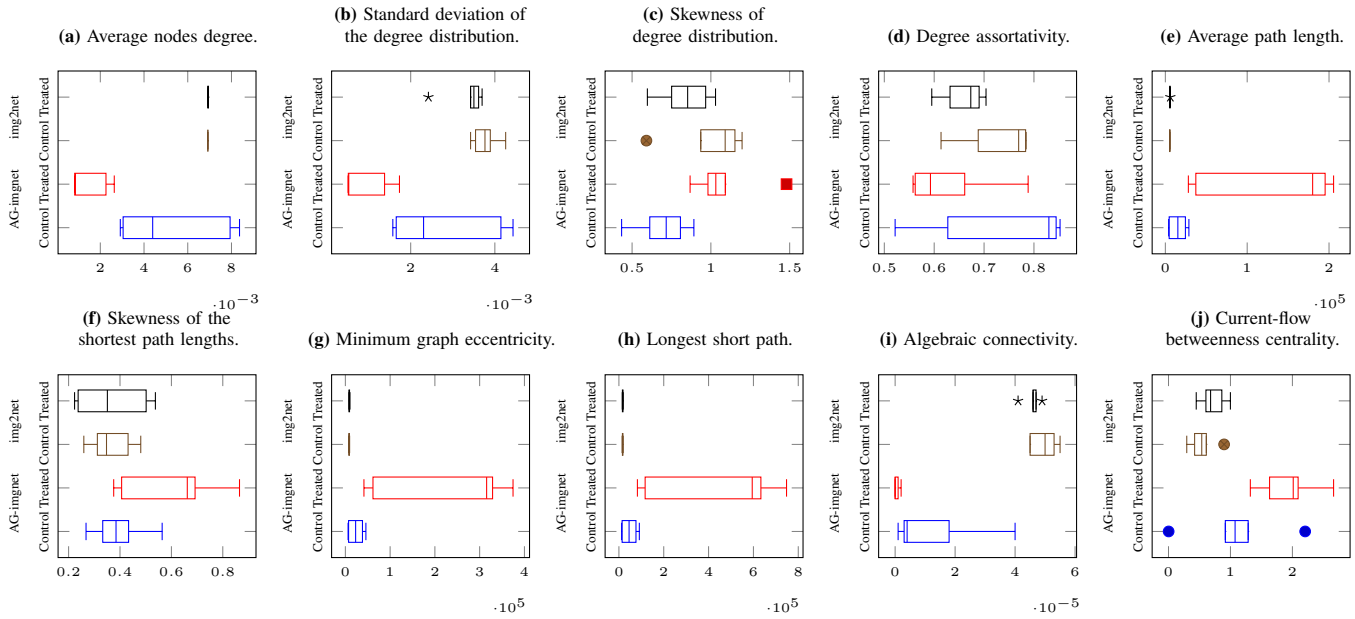


Figure 7: Comparison between the properties of treated and untreated cytoskeleton networks measured using AG-img2net and img2net [4], [6]. img2net: in the top two boxplots of each measured property plot; AG-img2net: in the bottom two boxplots of each measured property plot. (a) Average nodes degree. (b) Standard deviation of nodes distances. (c) Skewness of degree distribution. (d) Assorativity. (e) Average path length. (f) Skewness of the shortest path lengths. (g) Minimum graph eccentricity (radius). (h) Longest short path (diameter). (i) Algebraic connectivity. (j) Skewness of betweenness centrality of nodes.

achieves significant results and speed-up over other state-of-the-art approaches.

VII. Future Work

In order to further advance the current achievements of this study, a few future directions to follow are:

- The system should be able to capture time series in 2D and 3D images.
- The validation of this system and its results should be carried out on a greater variety of biological networks.
- In addition, a parallel approach needs to be developed in order to improve performance.

VIII. Implementation

We implemented a pointer-less quadtree partitioning based on [16], [24] and performed the graph analysis based on img2net [4], [6] and the Networkx Python library [28], using the same graph properties in img2net (as described in Table IV). We used the following packages to implement our work: Python 2.7.3, SciPy, NumPy, nested_dict 1.61, Cython and NetworkX.

Acknowledgments

Haifa Alhasson is supported by Saudi Arabian Ministry of Higher Education Doctoral Scholarship. The work in this paper was supported by an academic grant from the The Royal Society (UK; RF080232).

References

[1] M. Dirnberger, T. Kehl, and A. Neumann, "NEFI: Network Extraction From Images," *Scientific Reports*, vol. 5, p. 15669, 2015.

[2] B. Obara, M. Fricker, and V. Grau, "A bioimage informatics approach to automatically extract complex fungal networks," *Bioinformatics*, vol. 28, no. 18, pp. 2374–2381, 2012.

[3] D. Monroe, "Focus: Why leaves aren't trees," *Physics*, vol. 25, p. 4, 2010.

[4] D. Breuer, A. Ivakov, A. Sampathkumar, F. Hollandt, S. Persson, and Z. Nikoloski, "Quantitative analysis of the plant cytoskeleton reveal underlying organizational principles," *Journal of The Royal Society Interface*, vol. 11, no. 97, pp. 362–372, 2014.

[5] W. Baumgarten, "Detection, extraction, and analysis of the vein network of the slime mould physarum polycephalum," *Journal of Computational Interdisciplinary Sciences*, vol. 1, no. 3, p. 241, 2010.

[6] D. Breuer and Z. Nikoloski, "img2net: automated network-based analysis of imaged phenotypes," *Bioinformatics*, p. btu503, 2014.

[7] A. A. Hagberg, D. A. Schult, and P. J. Swart, "Exploring network structure, dynamics, and function using NetworkX," in *Python in Science*, Pasadena, CA USA, Aug 2008, pp. 11–15.

[8] E. Steinthorsson and D. Modiano, "Advanced methodology for simulation of complex flows using structured grid systems," vol. 1, pp. 697–710, 1995.

[9] M. Vajteršic, P. Zinterhof, and R. Trobec, *Overview–Parallel Computing: Numerics, Applications, and Trends*. Springer, 2009.

[10] C. Burstedde, L. C. Wilcox, and O. Ghattas, "p4est: Scalable algorithms for parallel adaptive mesh refinement on forests of octrees," *SIAM Journal on Scientific Computing*, vol. 33, no. 3, pp. 1103–1133, 2011.

[11] R. A. Finkel and J. L. Bentley, "Quadtrees a data structure for retrieval on composite keys," *Acta informatica*, vol. 4, no. 1, pp. 1–9, 1974.

[12] H. Samet, *Applications of spatial data structures: Computer graphics, image processing, and GIS*. Addison-Wesley Longman Publishing Co., Inc., Jan 1990.

[13] D. P. Mehta and S. Sahni, *Handbook of data structures and applications*, ser. Computer and information science series. CRC Press, 2004.

[14] H. Sundar, R. S. Sampath, and G. Biros, "Bottom-up construction and 2:1 balance refinement of linear octrees in parallel," *SIAM Journal on Scientific Computing*, vol. 30, no. 5, pp. 2675–2708, 2008.

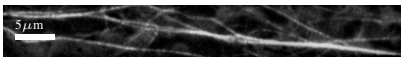
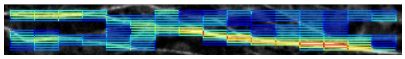
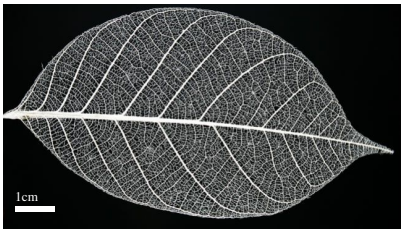
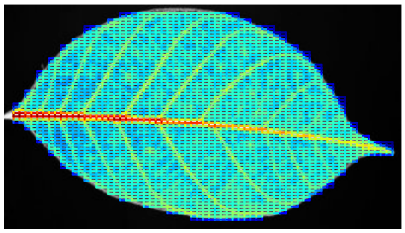
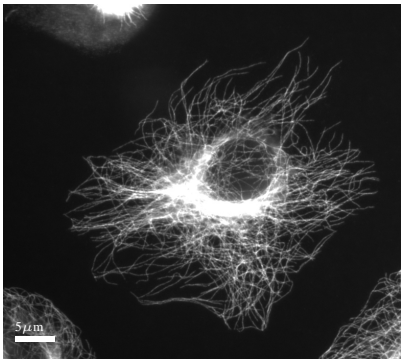
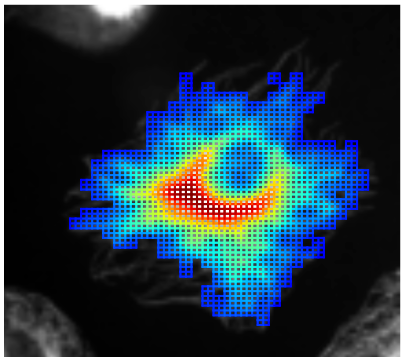
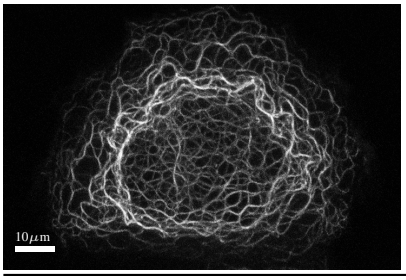
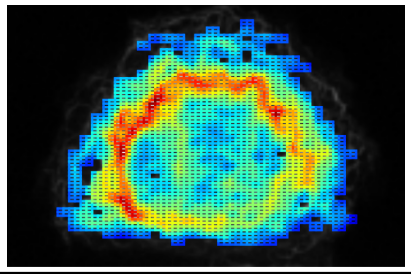
[15] S. N. Dorogovtsev and J. F. Mendes, *Evolution of networks: From biological nets to the Internet and WWW*. OUP Oxford, 2013.

[16] I. Gargantini, "An effective way to represent quadtrees," *Communications of the ACM*, vol. 25, no. 12, pp. 905–910, 1982.

[17] T. Lewiner, V. Mello, A. Peixoto, S. Pesco, and H. Lopes, "Fast generation of pointerless octree duals," in *Computer Graphics Forum*, vol. 29, no. 5, 2010, pp. 1661–1669.

[18] S. Lefebvre and H. Hoppe, "Perfect spatial hashing," in *ACM Transactions on Graphics (TOG)*, vol. 25, no. 3, 2006, pp. 579–588.

Table II: Results obtained by the AG-img2net approach when applied to the selected biological networks.

Images	Details	Results
	**Type:** Cytoskeleton in plant cells. **Provided by:** [6]	
	**Type:** Vein leaf network. **Provided by:** Prof. Mark Fricker, Department of Plant Sciences, Oxford University.	
	**Type:** Microtubules in cells. **Provided by:** [27]	
	**Type:** Cytoskeletal network in a skin cell. **Provided by:** Prof. Dr. med. Rudolf Leube, Institute of Molecular and Cellular Anatomy, RWTH Aachen University, Germany.	

[19] A. W. Reza, C. Eswaran, and S. Hati, "Diabetic retinopathy: A quadtree based blood vessel detection algorithm using rgb components in fundus images," *Journal of medical systems*, vol. 32, no. 2, pp. 147–155, 2008.

[20] N. S. Rani and A. Gopi, "A quad tree based binarization approach to improve quality of degraded document images," *International Journal of Computer Science Engineering*, vol. 3, no. 1, 2014.

[21] N. Otsu, "A threshold selection method from gray-level histograms," *Automatica*, vol. 11, no. 285-296, pp. 23–27, 1975.

[22] W. Niblack, *An Introduction to Digital Image Processing*. Birkeroed, Denmark, Denmark: Strandberg Publishing Company, 1985.

[23] J. Sauvola and M. Pietikäinen, "Adaptive document image binarization," *Pattern recognition*, vol. 33, no. 2, pp. 225–236, 2000.

[24] M. G. Choi, E. Ju, J.-W. Chang, J. Lee, and Y. J. Kim, "Linkless octree using multi-level perfect hashing," in *Computer Graphics Forum*, vol. 28, no. 7, 2009, pp. 1773–1780.

[25] M. de Berg, M. van Krefeld, M. Overmars, and O. Schwarzkopf, "Computational geometry: Algorithms and applications," 2000.

[26] R. Wedowski, A. R. Farooq, L. N. Smith, and M. L. Smith, "High speed, multi-scale tracing of curvilinear features with automated scale selection and enhanced orientation computation," in *High Performance Computing and Simulation (HPCS), 2010 International Conference on*, 2010, pp. 410–417.

[27] (2006). [Online]. Available: https://commons.wikimedia.org/wiki/File:Btub.jpg

[28] I. Neri, N. Kern, and A. Parmeggiani, "Modeling cytoskeletal traffic: an interplay between passive diffusion and active transport," *Physical review letters*, vol. 110, no. 9, p. 098102, 2013.

[29] D. West, *Introduction to Graph Theory*, ser. Featured Titles for Graph Theory Series. Prentice Hall, 2001.

[30] W. Ellens, F. Spieksma, P. Van Mieghem, A. Jamakovic, and R. Kooij, "Effective graph resistance," *Linear Algebra and its Applications*, vol. 435, no. 10, pp. 2491–2506, 2011.

[31] A. Kelmans, "On the Laplacian spectrum of (α, ω)-Graphs," *European Journal of Combinatorics*, vol. 23, no. 6, pp. 673–682, 2002.

[32] M. E. Newman, "The structure and function of complex networks," *SIAM review*, vol. 45, no. 2, pp. 167–256, 2003.

[33] X. Zhu, M. Gerstein, and M. Snyder, "Getting connected: analysis and principles of biological networks," *Genes & development*, vol. 21, no. 9, pp. 1010–1024, 2007.

[34] R. J. Wilson, "Introduction to graph theory," *Discrete Mathematics*, vol. 37, no. 1, p. 133, 1981.

Table III: Comparison in performance between our network analysis system AG-img2net and img2net [6].

Approach	Image samples	Size [pixels]	Nodes	Edges	Time of constructing grid [s]	Time of analysing network [s]	Total time [s]
img2net	Cytoskeleton in plant cells [6]	50 ×390	729	1716	31.3	18	49.3
	Microtubules in cells [27]	109×97	1152	2236	42.2	36.8	49
	Cytoskeletal in skin cells	97×174	1568	3055	87.15	80.3	167.45
	Vein leaf network	322×188	3600	7075	392.9	488.4	881.3
AG-img2net	Cytoskeleton in plant cells [6]	50 ×390	784	1406	4.4	17.9	22.3
	Microtubules in cells [27]	109×97	1654	3160	14.4	75.3	89.7
	Cytoskeletal in skin cells	97×174	2328	4474	26.6	160	186.6
	Vein leaf network	322×188	2873	5616	68.8	275.9	344.7

Table IV: Overview of NetworkX graph measurements used to measure network topological properties.

Measurement	Description
Mean[degree]	Defined as the average node degree. Node degree refers to the number of links attached to a node. Often used as a measure of connectedness of networks, especially the frequency distribution of node degrees.
Median[degree]	The middle number of sorted node degrees. If the mean is equal to the median and the mode, then the distribution is uni-modal.
Sd[degree]	The measure used to quantify the amount of variation of a set of node degrees. The standard deviation of the degree distribution [29] is used to distinguish the spatial heterogeneity of the distribution of actin structures. In our case, the node degree reflects the intensity of underlying cytoskeleton, thus it is used to quantify the spatial heterogeneity of the distribution of intensities in the underlying cytoskeleton images [6].
Skewness[degree]	A measure of symmetry [29] or, to be more precise, a measure used to capture the asymmetry of the probability distribution of node degrees about its mean. If the mean is equal to the median and the distribution has a zero skewness, this distribution is symmetric.
Clustering coefficient	A measure of the degree to which nodes in a graph tend to cluster together [15].
Degree assortativity	A measure of the correlation of degrees of neighbouring nodes, reflecting how similar the connections are in the graph with respect to the node degree [30].
Mean[distance]	The average value of the shortest path lengths between all nodes in a weighted graph, quantifying the overall transport efficiency of the network. If the network has smaller average shortest paths, meaning a similar physical extent, it has a more efficient transport system [31].
Sd[distance]	The measure used to quantify the amount of variation of the shortest path lengths between each two nodes in a weighted graph.
Skewness[distance]	The measure used to capture the asymmetry of the shortest path lengths between every two nodes in a weighted graph.
Radius	Defined as minimum graph eccentricity of any graph vertex in a graph, where the eccentricity $ecc(v)$ of v in graph G is the greatest distance from v to any other node [32].
Diameter	Defined as the longest short path, or it's the greatest distance between any pair of vertices in the graph. [33].
Effective resistance	Defined as a function of the Laplacian eigenvalues of the graph, calculated as a sum of the effective resistances over all pairs of vertices. More informally, the effective resistance between two vertices of a network, assuming that a network is seen as an electrical circuit, can be calculated by the well-known series and parallel manipulations [34].
Algebraic connectivity	Defined as the second smallest eigenvalue of the Laplacian matrix [31]. This reflects the level of connectivity of the graph. Therefore, it could be defined as the minimum collection of weights of edges that necessarily needs to be removed to disconnect the network [6].
Mean[betweenness]	The fraction of all shortest paths connecting a pair of vertices that pass through a given vertex, averaged over all nodes. Or it could be defined as the average of the random-walk betweenness centrality [32].
Sd[betweenness]	A measure to describe how the central the nodes are in a graph. A high mean betweenness centrality means the nodes clustered together and it gives low mean betweenness when the graph is a spread out [29].
Skewness[betweenness]	It is known as current-flow betweenness centrality used to capture the asymmetry of probability distribution of nodes centrality about its mean [6].
Angle i	The sum of the edge weights when there is a contribution of an imaginary rod placed on the image to the weight of edges with i degree where $i = 0, 45, 60, 90, 120, 135$ [6].
Angle ratio $00 - 90$	The ratio between edge weight for angle 0 and angle 90 [6].